飞龙的程序员书单

wizardforcel

Published
with GitBook

目錄

介紹	0
编程语言	1
前端	2
Linux	3
算法	4
组原、OS、网络	5
思想、工程、架构、职业发展	6
其它	7
参考列表	8

飞龙的程序员书单

编程语言

C

C程序设计语言

丢掉老谭的那本书吧，这本书才是最好的入门书。这本书两百多页，语言简洁，但又覆盖到了C语言的每个方面。然而这本书不仅仅讲解C语言，还附带讲解了二分查找、快速排序、二叉树、哈希表这些重要的数据结构和算法。甚至为了解释让人头疼的复杂声明，写了一个递归下降的parser。就算你不是拿此书入门的，也可以放在手头当作参考书来用。

C和指针

可以说，选择了C语言就选择了指针，C语言和其它语言不同的一点就是，C语言避不开指针。这本书的确详细讲了高阶指针的复杂逻辑，解释了指针与数组和字符串的关系，但它又不是一本仅仅讲指针的书。Pointer也可以理解为建议或者提示，书中也的确给了一些C语言编程的技巧和提示，这才是此书的本意。

C语言程序设计现代方法

C语言的语法相对简单，所以书非常薄；然而，语法简单就意味着你要手写更多的代码来实现本该由编译器实现的特性，所以在实践中用好C语言极其困难。这是一本相对较新的书。我推荐此书的原因是，它的重点是“现代”的C语言。此书强调现代的软件开发，不再强调作坊式的手工优化。书中有大量的实例，关于如何在实践中用好C语言，让C语言在实际开发中发挥作用，而不是只是停留在考试与刷题上面。

C++

C++最近处在一个不稳定的状态，很多新的特性和标准库设施的加入使得开发者需要重新熟悉和审视一些东西。现代的C++开发，要充分利用新的标准库，尽可能摒弃继承自C的糟粕。所以这方面的书务必要看最新的。

入门

C++ Primer

如果只能选择一本C++入门书，那么绝对是它了。第五版根据C++11标准修订，将STL、string和智能指针在相应的章节提前讲解，并且一开始就默认使用vector和string，符合C++现代编程规范。

完美C++

传说中的Absolute C++，其翻译从第二版直接跨越到了第五版。我记忆尤其深刻的一句话就是，“掌握了变量的使用，就入门了编程”。如果你阅读C++ Primer觉得吃力，不妨看看这本书。

进阶

Effective 系列

- Effective C++
- More Effective C++
- Effective STL
- Effective Modern C++

所谓“最佳实践”，就是前人在编写代码时过程中思考并总结出的一些东西。我们的时间有限，要想提升项目质量，就要读effective系列的书。尤其是对于没有什么开发经验的应届生，不可不读。

深入理解C++11

此书是C++标准委员会代表和IBM XL编译器中国开发团队共同撰写，全面解析C++11新增特性。想要写出符合现代C++规范的代码，不可不了解C++11的新特性。

C++程序设计原理与实践

这本书不罗列枯燥无味的概念和知识，相反，每个特性都用了简短的例子来说明如何在实战中使用。书中还给出了两个综合的示例，一个是使用词法分析编写一个解析表达式的计算器；另外一个是使用面向对象思想如何构造一个复杂的GUI程序。有趣的是，这本书毫无提及到指针，说明了指针在C++中其实不是必要的。但是总是把对象放在栈上也不方便，现在智能指针出来之后，我们完全可以绕过原生指针了。

参考

C++标准库

此书第一版是2003年出版的，对应C++03，为了迎接C++11，13年来首次再版。由于标准库加入了许多新的设施，此书也从800页增加到了1100多页。

C++程序设计语言

这本书更像是C++语法及特性的参考，拿来入门不太合适。适合放在手头参阅。另外这本书已经有了国际中文版，gotop出版的。

Java

入门

Java编程思想

Java是完全面向对象的，学习Java就要先入为主有对象的概念，这一点来说，这本书做的不错。这本书不会过多提及语言特性，而是顺带讲解面向对象的一些思想，引人深思。

Java核心技术 系列

- Java核心技术 卷一
- Java核心技术 卷二

这两本书的特点是比较简略，并带有同C++的比较，便于C++程序员迁移。卷一除了基本语法之外，还讲解了容器和并发这两个现代编程语言中重要的概念。图形编程部分可以跳读。卷二虽然是高级特性，但是其中的io、文件系统、正则表达式、网络、jdbc也是日常编程中非常常用的组件，不可不看。个人认为这套书的组织比较混乱，如果把第二本的那三章重要的部分放进第一本，第一本的图形编程放到第二版，会更好一点。

进阶

Effective java

深入理解Java虚拟机

要想多快好省编写java程序，不可不了解jvm。

Java特种兵（上册）

由于是进阶书，这本书并不会详细讲解每一个知识点，而是对实际开发中常用的组件和一些坑做了梳理。另外此书还对编程中一些共性的问题，比如架构和实际开发的体会做了一些阐述。

写给大忙人看的Java SE 8

由于java8推出之时，java核心技术没有及时更新，这本书算是对java核心技术的很好的补充。

参考

Java技术手册

C#

入门

C#入门经典

C# 高级编程

C# 图解教程

精通 C#

别被书名吓到，这四本其实都是入门用的书籍。挑一本看就行了，或者你喜欢看MSDN的编程指南也可以。

进阶

深入理解C#

这本书是介绍c#新特性的书，按照历史发展顺序，从.net1.0到5.0对新特性做了解释和使用示例。

CLR via C#

.net平台之于c#相当于jvm之于java，重要性不言而喻。此书是深入了解c#底层.net平台的最佳读物。

Effective 系列

- Effective C#
- More Effective C#

参考

这部分的书籍不做推荐，因为MSDN的.net类库参考已经很翔实了，可以直接去翻阅。

Python

说实话，Python的免费教程很多，而且质量也不错，涵盖的范围远比书籍要宽，更新也比较及时。这里我只是把看过而且觉得不错的书列出来。

Python核心编程

Python基础教程

“笨办法”学Python

这本书是开放的，在很多文档站上都可以找到，比如这里。

Effective Python

前端

作为一个主打后端的开发者，还是应该了解一下前端的。这里只是列出一些涉及到html、css和js的书，框架还是直接读文档比较好，这些东西更新迅速，书一旦出版就过时了。

html

入门

HTML5与CSS3基础教程

进阶

HTML5秘籍

HTML5程序设计

参考

HTML5权威指南

CSS

HTML5与CSS3基础教程

CSS权威指南

Javascript

入门

JavaScript基础教程

JavaScript高级程序设计

进阶

你不知道的JavaScript（上卷）

这本书简称“ydkjs”，英文版是开放的，在github上面托管。这本书深入分析了一些核心特性的原理，对于程序员了解语言背后的机制很有帮助。

JavaScript DOM编程艺术

ECMAScript 6入门

阮一峰老师所写的开放书籍，全面介绍ECMAScript 6新引入的语法特性。此书为进阶级，适合已有一定JavaScript语言基础的读者，用来了解这门语言的最新发展；也可当作参考手册，查寻新增的语法点。在阮老师的网站上可以在线阅读。

参考

JavaScript 权威指南

jQuery

精通jQuery

此外，W3School和MDN上也有html、css、js和jq的详尽参考手册。

Linux

关于Linux或者Unix的书单罗列在这里。之所以不把他们划分进OS里面，是因为Linux已然成为了一个生态，并不是操作系统那么简单。

使用

鸟哥的Linux私房菜.基础学习篇

鸟哥的Linux私房菜：服务器架设篇

这两本书都是开放的，在这里可以在线阅读。

编程

Linux程序设计

UNIX环境高级编程

UNIX网络编程

- UNIX网络编程 卷1
- UNIX网络编程 卷2

学习C语言的话，只了解标准库API是不够的，还需要了解posix api，才能写出来可用的程序。

内核

Linux内核设计与实现

其它

UNIX编程艺术

数据结构、算法

入门向

啊哈!算法

这本书真心简洁易懂，dijkstra我是看课本怎么看也看不懂，最后看这本书才懂的。真心推荐。

大话数据结构

工程向

算法

- Java实现
- C实现
- C++实现

普林斯顿的算法课程教材，Coursera上面有配套的在线视频。这套书不仅有三种实现，而且每本书都有相应编程语言的实例，而不是伪代码。适合技术人员阅读。

数据结构与算法分析

- Java描述
- C描述
- C++描述

学术向

算法导论

这本书不应该叫导论，应该叫大全。所以适合放在进阶阶段阅读。这本书也比较偏证明，全是伪代码，所以我说它是学术向的，一般技术人员很难用到。如果实在啃不动的话，不要勉强自己，你可能不一定非要读这本书。

竞赛相关

算法竞赛入门经典

竞赛的题型高度模式化，和工程向的问题风格不一样。而且技术人员所需的算法无论是面试时遇到的，还是实际项目中碰到的，都没有那么难。

面试相关

剑指Offer

组原、OS、网络

组原

深入理解计算机系统

简单介绍一下，这本书包括组成原理和操作系统两大部分知识。第二、三章学完之后，逆向就算是入门了。国内的教材很少有拿汇编和C语言对比教学的书籍，这样的教学方法很实用。因为现在汇编很少用于写程序了，主要大量用于反编译；而C语言主要用于嵌入式开发，为了性能原因开发者要经常查看对应的汇编代码。然而这些知识只是个导论，要想真正踏入逆向的大门，还是要看看雪出的那几本书。

之后的第四、五章涉及了指令集和体系结构，讲的很浅显易懂，完全不需要数电的基础就能看懂。美中不足的一点是它拿y86作为教学工具。y86虽然比起x86简化了很多，但还是属于CISC，基本的指令类型有12种，很是令人头疼。此外，这一部分的lab涉及到了HDL，然而相关语法需要自己去看，也是略有不足。

操作系统部分没什么可挑剔的，进程、并发、IO与网络为了不那么抽象，特地采用posix api来做演示，是相当深入浅出的。当然，一本导论性质的书不可能讲的那么详细，需要的话还是要看《apue》。

另外，这本书好像遗漏了两个很重要的话题，组原部分少了总线，操作系统部分少了文件系统，所以还是应该看别的书把它补上。

计算机组成与设计

斯坦福校长写的书。如果你受不了csapp中y86的繁琐，来看这本吧。其中有一个简单的用于教学的mips，只需要记住和实现5种指令，简单吧。

OS

深入理解计算机系统

现代操作系统

一本中规中矩的OS教科书，主要涉及到了进程、并发与锁、IO、虚存、文件系统几个方面。

30天自制操作系统

和上面那本不同，这本书不涉及到一些操作系统设计上的基本概念，而是涉及到boot和内核的一些知识，教你从头开始实现一个操作系统，这是非常难能可贵的。

网络

计算机网络

计算机网络：自顶向下方法

这两本书前者是自底向上，后者是自顶向下。可以互相补充着看。另外不要再看谢希仁那本书了，那本书都是他让手下的研究生一章一章凑的，没有多大价值。

TCP/IP详解 卷1：协议

TCP/IP详解 卷2：实现

TCP/IP详解 卷3：TCP事务协议、HTTP、NNTP和UNIX域协议

作为TCP/IP的权威指南，这三本还是挺晦涩难懂的。

HTTP权威指南

作为一名开发者，我们其实只需要关注传输层和应用层的东西。了解HTTP，对于一个web开发者来说，是极其有利的。至少能理清请求的来龙去脉，避免写出来那种跳来跳去的实现。

图解HTTP

图解TCP/IP

如果上面那些大部头看不下去了，看看下面这两本，图解有时候真的有奇效。

思想、工程、架构、职业发展

思想

编程珠玑

编程珠玑，字字珠玑。这本书并不单纯讲解算法，而是由一系列实际问题切入，引导读者理解这些问题并学会解决方法，使用现有的算法，或者程序设计技巧解决问题。本书的特色是通过一些精心设计的有趣而又颇具指导意义的程序，对实用程序设计技巧及基本设计原则进行了透彻而睿智的描述，为复杂的编程问题提供了清晰而完备的解决思路。

代码之美

33章，有38位作者，每位作者贡献一章。在每章中的漂亮代码都是来自独特解决方案的发现，而这种发现是来源于作者超越既定边界的远见卓识，并且识别出被多数人忽视的需求以及找出令人叹为观止的问题解决方案。每位作者都将自己心目中对于“美丽的代码”的认识浓缩在一章当中，张力十足。38位大牛，每个人对代码之美都有自己独特的认识，现在一览无余的放在一起，对于热爱程序的每个人都不啻一场盛宴。

计算机程序的构造和解释

本书是lisp的入门书，也是讲授函数式语言思想的一本书。由于函数式语言特性，放在编程思想里也是可以的。这绝对不是一本初学者需要的书，但是是一本接触函数式编程就必须要读的书。十年前这本书出版了，十年后肯定会有机会需要读这本书。

程序员思维修炼

作为程序员，我们需要不断地学习——既要学习新技术，也要学习如何解决各种领域的问题。为了提升学习效率，我们需要学习如何学习。程序员的思维修炼正是这样一本讲如何学习的书，它集合了认知科学，神经学，以及行为理论的最新研究成果，并系统的介绍了大脑的工作机制。通过这本书，你将学会如何高效的使用自己的大脑，从而提高思考能力，改善学习效率。

架构

设计模式

这本书不用说了，gof book，设计模式相关书籍的鼻祖。任何讲设计模式的书，其实都离不开这23种设计模式。这本书不仅仅解释了每种设计模式的架构，还附上一个例子来讲解如何使用。

大话设计模式

如果上面那本看不下去的话，来看看这本。

企业应用架构模式

这本书主要讲解了企业应用（也就是大规模的web应用）所使用的架构风格，例如mvc，orm与事务。可以作为设计模式的补充。

架构之美

所谓好的架构，使用者感觉不到，但是用起来顺手。这本书让最优秀的设计师和架构师来描述他们选择的软件架构，剥开架构的各层，展示他们如何让软件做到实现功能、可靠、易用、高效率、可维护、可移植和优雅。

重构

如果说设计模式是涉及到类于类的关系，那这本书着重于类中方法的组织，和方法中代码的组织。这本书绝对不是仅仅是讲如何重构的书，它更像是一本编写可维护代码的指导手册。如何编写避免重构的代码才是本书的要义。

工程

代码大全

这本书涵盖了从项目建立、需求分析、编码、测试到交付的所有细节、最佳实践和坑，堪称软件工程的百科全书。既然前人已经帮你把坑都踩过了，有什么理由不去读它呢？

人月神话

作为软件工程的经典著作，《人月神话》的主要贡献是对软件开发过程的几个重要关键点，提出了独到的见解。主要包括：（1）外科手术式的团队组织；（2）核心概念要由很少的人来完成，保证概念的完整性；（3）必要的沟通；（4）保持适度的文档；（5）没有包治百病的银弹。

人件

《人件》是项目管理图书中的经典之作，和《人月神话》齐名。这本书专门讨论了软件开发和维护的团队管理问题，并向人们的传统认识提出了挑战。书中指出了知识型企业的核心是人，而不是技术。《人月神话》关注“软件开发”本身，《人件》则关注软件开发中的“人”。

番茄土豆工作法

这本书介绍了时下最流行的时间管理方法之一：番茄工作法。早在小学语文课本上，李大钊先生就教育我们，“学就学个踏实，玩就玩个痛快。”番茄法正是这样。这本书有助于提升拖延症人士的工作效率，让人受益良多。

高效程序员的45个习惯

职业发展

程序员修炼之道

这本书并不局限于特定的编程语言或框架，而是提出了一套切实可行的实效（Pragmatic）开发哲学，并通过程序设计，测试，编程工具，以及项目管理等方面的实例展示了如何应用这套开发哲学，从而使得程序员更加高效专业。有人把这本书称之为迷你版代码大全——代码大全给出了大量的优秀程序设计实践，偏向术；而程序员修炼之道给出了程序设计实践背后的思想，注重道。

程序员职业规划之道

这本书运用其独特的思维方式，启发程序员不能只注重技能上的提高，还要关注自己的职业发展。书中涉及新旧技术的取舍、技术与业务的关系、技术是要专精还是要广博等，相信这也是长久以来困扰你的问题。带着这些问题去阅读此书，定会受益良多。此外，本书中的每一章都包含一篇或几篇各领域成功人士的文章，让你直接了解他们如何规划自己的职业生涯。

程序员面试宝典

其它

逆向

加密与解密

C++反汇编与逆向分析技术揭秘

微软.NET程序的加密与解密

IDA Pro权威指南

Windows PE权威指南

Android软件安全与逆向分析

Android安全攻防权威指南

渗透

白帽子讲Web安全

Web前端黑客技术揭秘

XSS跨站脚本攻击剖析与防御

黑客攻防技术宝典：Web实战篇

Metasploit渗透测试指南

Web应用安全权威指南

渗透测试实践指南

编译

编译原理

自制编程语言

数据库

数据库系统概念

高性能MySQL

SQL反模式

Redis

Redis设计与实现

Redis实战

MongoDB

MongoDB权威指南

数据挖掘

数据挖掘：概念与技术

数据挖掘导论

数据挖掘十大算法

IT人文

这一部分是适合躺在床上看的书，2~3天即可看完一本。

黑客与画家

Joel说软件

软件随想录

世界是数字的

编程人生

浪潮之巅

软件故事

我是一只IT小小鸟

疯狂的程序员

[线上幽灵]

参考列表

- 豆瓣图书
- 程序员必读书单 1.0
- What is the single most influential book every programmer should read?